

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

**(19) World Intellectual Property Organization
International Bureau**



A standard linear barcode is located at the bottom of the page, spanning most of the width.

(43) International Publication Date
27 November 2003 (27.11.2003)

PCT

(10) International Publication Number
WO 03/098461 A1

(51) International Patent Classification⁷: G06F 15/177, (74) Agents: FLESHNER, Mark, L. et al.; Fleshner & Kim, 13/00 LLP, P.O. Box 221200, Chantilly, VA 20153-1200 (US).

(21) International Application Number: PCT/US03/13407

(22) International Filing Date: 14 May 2003 (14.05.2003)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:

60/379,757	14 May 2002 (14.05.2002)	US
60/426,341	15 November 2002 (15.11.2002)	US
10/321,722	18 December 2002 (18.12.2002)	US
60/436,996	31 December 2002 (31.12.2002)	US

(74) Agents: **FLESHNER, Mark, L.** et al.; Fleshner & Kim, LLP, P.O. Box 221200, Chantilly, VA 20153-1200 (US).

(81) **Designated States (national):** AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NI, NO, NZ, OM, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.

(84) **Designated States (regional):** ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PT, RO, SE, SI, SK, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

(71) Applicants and

(72) Inventors: POLCHA, Andrew, J. [US/US]; 12462

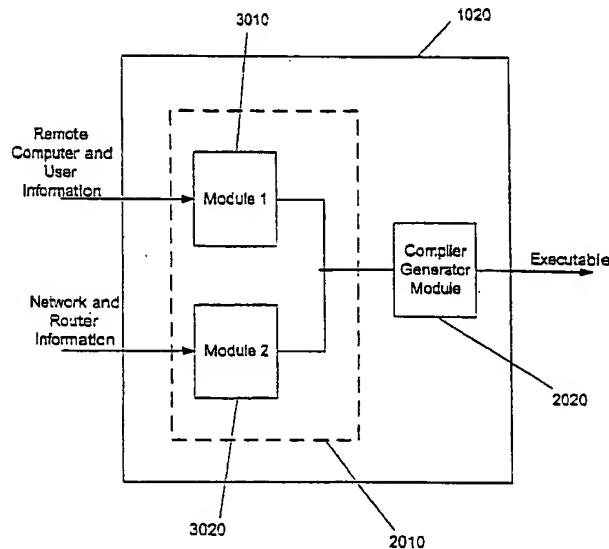
Cabin Spring Lane, Lovettsville, VA 20180 (US).

POLCHA, Michael, P. [US/US]; 12462 Cabin Spring Lane, Lovettsville, VA 20180 (US).

Published:
— with international search report

[Continued on next page]

(54) Title: SYSTEM AND METHOD FOR AUTOMATICALLY CONFIGURING REMOTE COMPUTER



WO 03/098461 A1

(57) Abstract: A system and method for automatically configuring a remote computer to establish a VPN connection with a network is disclosed. According to the system and method, a configuration generator (1020) determines information necessary for the remote computer to establish the VPN connection, and generate an executable file to be run the remote computer. When run on the remote computer, the executable file automatically updates the remote computer's configuration settings so as to enable it to establish the connection with the network. The information required by the configuration generator (1020) can either be extracted from a database automatically, or can be keyed in by a network administrator.

WO 03/098461 A1



For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

SYSTEM AND METHOD FOR AUTOMATICALLY CONFIGURING REMOTE COMPUTER

BACKGROUND OF THE INVENTION

Field of the Invention

5 The present invention relates to remote computer configuration, and more particularly to a system and method for automatically configuring a remote computer for establishing a virtual private network (VPN).

Background of the Related Art

10 A Virtual Private Network (VPN) is a secure communication channel established between computers over a shared public network. In most instances, the VPN connects a remote user's computer (sometimes referred to as the caller) with a local host computer (sometimes referred to as the callee) over the Internet. Thus, for example, a user's home computer can be connected to a local area network (LAN) at the user's place of employment. Before such a connection can be
15 established, however, the user's home computer and the LAN need to be properly configured for the remote connection.

Referring to Figure 1, a structure of a related art VPN is shown. Referring to Figure 1, a LAN includes at least one server 100 having various local user computers 100-1 ~ 100-N connected thereto. Additionally, the server 100 is coupled to a gateway/router 120 to access the
20 Internet 130. The gateway/router 120 allows for two-way communication between the server 100 and the Internet 130. Additionally, the gateway/router 120 can also be coupled to an Internet service provider (ISP) 150.

At least one remote computer 140-1 is also coupled to the Internet 130. The remote user
25 140-1 can connect to the Internet 130 using its own ISP 160-1. When the remote user 140-1 and the gateway/router 120 are properly configured, the remote computer 140-1 can establish a secure connection through the Internet 130 to the gateway/router 120. The VPN connection is thus established. Once the VPN connection is established, the remote computer 140-1 is seen by the server 100 as another local user connected to the server 100. For example, the remote computer 140-1 would be present in the network neighborhood administered by the server 100. Any number
30 of remote users could access the gateway/router 120 through the Internet 130, depending on the configuration of the gateway/router 120.

Before a VPN connection can be established between the remote computer 140-1 and the server 100, however, the remote computer 140-1 and the gateway/router 120 need to be properly configured. For example, the remote computer 140-1 needs to be given the proper network

address to the gateway/router 120, and needs to know how to access the server 100. For example, various client services must be installed and configured on the remote computer to access the server. The procedure for accessing the server 100 could include, for example, knowing proper user names and passwords.

5 A network administrator operating the LAN is typically required to configure both the gateway/router 120 and the remote computer 140-1. Because the gateway/router 120 is typically co-located with the LAN, and is also remotely accessible by using prescribed TELNET commands, the network administrator supporting the LAN can easily access the gateway/router 120 (being an Internet device) so as to properly configure it. However, most remote computers are, by definition, 10 not co-located with the LAN 100. Therefore, these computers are most often not remotely accessible to be remotely configured.

Accordingly, a problem with a related art VPN is that a network administrator needs to individually configure each remote computer so that the user will be able to access the LAN. This may require that the network administrator physically visit the remote location where the remote 15 computer is, or alternatively may require that a remote user bring the remote computer to the network administrator. This can be very inconvenient and cause delay in configuring the remote computer. Additionally, when establishing the connection between the remote computer 140-1 and the server 100, a remote user may be required to first access the remote user's Internet service provider, next access the gateway/router, and then access the LAN. Accordingly, there can be up 20 to three levels of authentication that need to be traversed by the remote user to gain access to the LAN. This can be difficult for some users to properly navigate.

One solution to this problem is to provide an HTML wizard that can provide instructions for a user to follow to configure a computer for remote access. However, because of the complexity of 25 properly configuring a remote computer for access, this method can prove ineffective. Additionally, even with the instructions, the process of configuring the remote computer can still be time-consuming. Finally, a user following the instructions of the wizard may still enter erroneous data, thus requiring a network administrator to first undo the mistake and then reconfigure the computer for remote access.

The above references are incorporated by reference herein where appropriate for 30 appropriate teachings of additional or alternative details, features and/or technical background.

SUMMARY OF THE INVENTION

An object of the invention is to solve at least the above problems and/or disadvantages and to provide at least the advantages described hereinafter.

It is another object of the present invention to provide a system and method for configuring a remote computer for access to a VPN by generating an executable code to be run on the remote computer that configures the remote computer.

It is another object of the present invention to provide a system and method for automatically configuring a remote computer to access a VPN by e-mailing an executable program to the remote user which, when executed, properly sets the required parameters to establish the VPN connection.

It is another object of the present invention to provide a system and method for automatically configuring a remote computer to access a VPN by generating an executable program that configures a computer to access an ISP, access a gateway/router, and login to a LAN by selecting a single program icon.

It is another object of the present invention to provide a system and method for remotely configuring an operating system of a remote computer by generating an executable file that, when executed, manipulates registry entries of the remote computer according to requirements of the user of the remote computer.

According to the preferred embodiment, a compiler module is provided to generate an executable program that can be run on a remote computer to configure the remote computer to establish a VPN connection to the host computer. The compiler module uses various information regarding the local/host computer to generate the executable file that will configure the remote computer. For example, the compiler module uses information on the WAN profile of the network/LAN, the primary and secondary DNS of the local computer, and the primary and secondary WINS address of the local computer. Additionally, the compiler module uses the type of network configuration (client services), such as Microsoft or Novell, as well as the configuration of the server (for example, as domain controller or server). Next, the gateway address of the local computer is also provided to the compiler module.

In the preferred embodiment, no information regarding the remote computer is necessary to generate the executable file. Specifically, the executable file will first determine the necessary information regarding the remote computer when it is executed on the remote computer, and then will use that information to properly configure the remote computer. However, it would also be possible to provide the configuration module with information regarding the operating system that the remote user has installed on the remote computer. By doing this, it may be possible to streamline the executable file that will eventually be delivered to the remote computer. That is, the executable file will already be tailored for the remote computer.

It is to be understood that any method can be used to provide the necessary information to the compiler module. In one embodiment, a system information front end module is provided to prompt the network administrator to provide the necessary information. The system information module then saves this information in a prescribed format to be used by the compiler module. In 5 another embodiment, the necessary information is extracted from a database into which the information has previously been stored.

Using the information, the compiler module generates an executable file including all of the parameters in a prescribed format. These parameters include client, adapter, protocol, and service. When run on a remote computer, the executable program configures the operating system with the 10 settings necessary to establish a VPN connection to the local/host network. Thus, it loads the client, adapter, protocol, and service. According to the preferred embodiment, the protocol is TCP/IP and the services file and printer sharing, which is turned on. In the preferred embodiment, the RAS phonebook of the Windows operating system is configured. Alternatively, registry entries can be directly manipulated by this executable program. The executable file can either be e-mailed 15 to the remote user or can be downloaded from a Web site. It should be understood that any method of providing this information to the remote user could be used, such as saving it to a disk that is sent to the remote user. To provide an additional layer of security, the executable file could be password protected such that the executable program would not run on the remote computer unless a prescribed password is provided upon executing the program. By doing this, if the 20 executable program is intercepted by an unintended recipient, the unintended recipient cannot gain access to the network because the executable program will not run without the password.

Additional information regarding the user can also be provided depending on the level of service that is intended to be provided to the user. For example, an automatic dialing program can be implemented in the executable program using the user's login and password information. 25 Moreover, automatic login information for the network can also be provided. When this additional user information is provided, after the executable configuration program is run on the remote computer, the remote user is given a single program to run (that is, a single icon to click), which will connect to and login to the host network. This single program is preferably a dialer program. Alternatively, the compiler module could generate both a configuration program and a separate 30 dialer program.

Additional advantages, objects, and features of the invention will be set forth in part in the description which follows and in part will become apparent to those having ordinary skill in the art upon examination of the following or may be learned from practice of the invention. The objects

and advantages of the invention may be realized and attained as particularly pointed out in the appended claims.

BRIEF DESCRIPTION OF THE DRAWINGS

- 5 The invention will be described in detail with reference to the following drawings in which like reference numerals refer to like elements wherein:
- Figure 1 shows a related art network configuration.
- Figure 2 illustrates a system for generating the executable remote configuration file according to the preferred embodiment.
- 10 Figure 3 illustrates a configuration program generator according to the preferred embodiment.
- Figure 4 illustrates the information collection module according to a first embodiment.
- Figure 5 illustrates additional details of the information collection module according to the first embodiment.
- 15 Figure 6 illustrates the user configuration tab of the information collection module according to the first embodiment.
- Figure 7 illustrates a construction of the database that is preferably used with the second embodiment of the information collection module.
- 20 Figure 8 illustrates a method of providing information to the information collection module according to the first embodiment.
- Figure 9 illustrates a method of providing WAN profile information to the information collection module according to the first embodiment.
- Figure 10 illustrates a method of providing remote user information to the information collection module according to the first embodiment.
- 25 Figure 11 illustrates a USB storage device including the executable remote configuration file according to the preferred embodiment.

DETAILED DESCRIPTION OF PREFERRED EMBODIMENTS

- The system and method of automatically configuring a remote computer to establish a VPN connection to a local computer preferably collects information about the remote computer, collects information about the local area network (LAN) and its associated gateway/router, and collects information regarding login and authentication procedures for the LAN and/or gateway/router. Using this information, at least one executable file is preferably generated for execution on the remote computer. The executable file is designed to configure the remote computer to establish a

VPN connection to the LAN through the gateway/router. In the preferred embodiment, the executable file manipulates the remote computer's registry information, such as the Remote Access Service (RAS). Additionally, in the preferred embodiment, the VPN connection is a client to gateway connection. That is, the remote computer is able to itself connect to the gateway of the local computer.

In order to collect the information needed to generate the executable file, the preferred embodiment uses a configuration program generator, which can be hardware, firmware, software, or any combination thereof. Preferably, the configuration program generator is made up of at least one or more software modules. Thus, referring to Figure 3, the configuration program generator preferably includes at least two modules. A first module collects information regarding the local computer (for example, the LAN server), the remote computer (for example, a PC), and user information for the operator of the remote computer. This module is referred to as the information collection module 2010.

A second module uses the information collected by the system information module to generate the executable file. This module is referred to as the compiler module 2020. The executable file is then preferably deployed by any manner, for example by e-mail, to the remote computer for execution.

As stated above, the configuration program generator is preferably software that would be run on a computer operated by the network administrator. This computer may be the server, or can be any other computer operated by the administrator. Additionally, the configuration program generator software could have any type of graphic user interface, or none at all. For example, in operation, the configuration program generator could be displayed as one or more dialog boxes. The dialog boxes could prompt the network administrator for information, or can be pre-populated with relevant information if the configuration program generator is associated with a database that has the necessary information for each potential remote user. Alternatively, if the configuration program generator is associated with such a database, it would be possible to implement the software with no graphical user interface.

Thus, in one embodiment, the network administrator could be provided with a great deal of control over the configuration program generator by having relevant information automatically retrieved from the database, and information input fields in various dialog boxes automatically populated with the retrieved information. The retrieved information would be information necessary to generate the executable file for any given remote user. In this case, one or more dialog boxes could be initially presented having blank/empty text fields corresponding to the various information necessary to establish the VPN connection. Upon receiving an instruction from the network

administrator, the relevant information could be retrieved from the database and directly inserted into the corresponding text boxes. The network administrator would thus have the opportunity to review and/or change the information in the text boxes prior to causing the configuration program generator to generate the executable file.

5 Alternatively, if there is no associated database, the configuration program generator could prompt an administrator to manually enter the required information into the corresponding text boxes. In this case, similar dialog boxes would be presented having blank text boxes. The network administrator would then enter the necessary information manually. In this embodiment, the configuration program generator could be further configured to provide the manually entered
10 information to a database for later use, in addition to generating the executable file.

In the preferred embodiment, however, the information collection module of the configuration program generator would transparently retrieve the necessary information directly from the database and pass it to the compiler module, without first displaying the information. The compiler module would then uses information to generate the executable file. A network
15 administrator may optionally be given the opportunity to confirm the information retrieved from the database prior to generating the executable file, or to modify the information.

Regardless of how the information is provided, there is certain network information that is necessary to establish the VPN connection. This network information includes, for example, a WAN number, one or two DNS numbers, one or two WINS numbers, and the gateway IP address.

20 Using the network information (either provided manually by the administrator or automatically extracted from the database), the configuration program generator preferably generates an executable program that can be run on the remote computer. The executable program, when run on the remote computer, would configure the remote computer so that it can establish a VPN connection with the local computer network. It is preferable that the executable
25 configuration program would update the registry settings of the remote computer or update the Windows settings. Depending on the operating system employed, various aspects of the registry and/or operating system data would be updated. Additionally, in a preferred embodiment, the executable file is encrypted prior to being deployed to the remote computer. In order to execute the executable file, the user of the remote computer would need to know the key to unlock the program.
30 This would prevent unauthorized users from executing the program, and provides a layer of security.

Referring to Figure 2, a conceptual view of the system for generating the executable remote configuration file is shown. The system preferably includes a computer/processor 1010 for executing commands to convert remote user information/network information into the executable

file. The computer/processor 1010 preferably has loaded thereon the configuration program generator 1020, which is preferably a software module configured to receive the remote user information and/or network information and generate the executable remote configuration file.

The software may be written in standard programming language, compiled and linked into 5 an executable program. For example, the software could be written in JavaScript, C++, PERL, and CGI. In the preferred embodiment, the software is one of an ActiveX or and OCX module, which can be linked to an application program such as Microsoft Access.

Additionally, it is possible to use scripting to cause the remote computer to execute certain 10 procedures. If scripting is used, the script first passes to an interpreter of the remote computer. The interpreter, preferably Windows, executes the software script into a readable medium. It should be understood that any other interpreter could be used, including UNIX, LINUX, or HTML. Moreover, it should be understood that it is possible to write code to emulate certain mouse clicks and keystrokes. For example, to configure a remote computer having Windows 98, it may be beneficial 15 to generate the executable file to emulate keystrokes and mouse clicks required to configure the system, instead of directly writing to the registry.

Additional details of how the executable file is generated will be described in a later section.

Referring to Figure 3, additional detail of the configuration program generator 1020 is shown. The configuration program generator 1020 preferably includes an information collection 20 module 2010 and a compiler module 2020.

The information collection module 2010 can be any software used to collect information needed by the compiler module to generate the executable remote configuration file. For example, in a very basic configuration, the information collection module could be an ASCII text comma separated document. In that case, the compiler module 2020 would preferably parse the 25 information in the ASCII document to retrieve the necessary information for generating the executable remote configuration file.

Alternatively, the information collection module 2010 could be a database with or without a filter. Preferably, using the filter, only relevant information for a prescribed user would be provided from the database to the compiler module 2020. Alternatively, information for several users could 30 be provided to compiler module 2020, so as to allow the compiler module 2020 to generate a plurality of executable files for each of the respective users. In either case, a network administrator could additionally review the information extracted from the database prior to transferring it to the compiler module 2020. Moreover, the network administrator could make changes to the data at that time. It should be understood that any method of compiling the database can be used.

Thus, in a first embodiment of the information collection module 2010, the network administrator enters the necessary information manually. Using this embodiment, no database is necessary. Rather, for each remote user for which a VPN connection will be configured, the relevant information is prided directly to the information collection module 2010 when the network 5 administrator needs to generate the executable file. In a second embodiment, on the other hand, a database is required. The network administrator would preferably select an appropriate record from the database. The record would preferably includes all of the information necessary to generate the executable files. The network administrator would then simply run the compiler module directly from the database to generate the executable file based on the selected record. In 10 this case, the information collection module 2010 could optionally include a verification module, displaying the information extracted from the database for the network administrator to verify prior to generating the executable file.

Regardless of the embodiment of the information collection module 2010, the compiler module 2020 uses the information collected to generate the executable program to be run on the 15 remote computer. Additional details of the information collection module 2010 and the compiler module 2020 are provided hereinbelow.

Referring to Figure 4, additional detail of the information collection module 2010 according to the first embodiment is provided. As shown therein, the information collection module 2010 is preferably comprised of two sub-modules 3010, 3020. The first sub-module is a remote user 20 information collection module 3010. The second sub-module is a network information collection module 3020. A network administrator would preferably manually enter information into each of these modules, according to one embodiment of the invention.

Each module of the information collection module 2010 is preferably displayed on a user interface screen (for example, a computer monitor) as a dialog box having prompts for the required 25 information. The network administrator would preferably enter the required information into the associated text box, or select the appropriate radio buttons, and then apply the settings. Upon applying the settings, the information collection module 2010 would format the selected information so as to be usable by the compiler module 2020. It should be understood that the dialog box could be formatted in any manner, and could be partitioned into any number of sections and/or sub-screens. Each section would preferably be identified by an associated tab, which could be selected 30 by the network administrator to access that section of the dialog box.

The remote user information collection module 3010 preferably collects information about the remote user and the remote user's computer. Specifically, it can preferably collect information concerning what operating system the remote user is running on the remote computer, as well as

password and logon information for the remote user. By way of example, the password/login information can be for either a dial-up adapter, the LAN to which the VPN will connect, or both.

The network information collection module 3020 preferably collects information about the local area network (LAN) to which the VPN connection will be established. This information is necessary for the remote user to negotiate to the router that is coupled to the LAN, as well as negotiate through the router and/or gateway to access the LAN. Additionally, further information regarding the names of the computers on the LAN should be provided, as well as the location of the domain name service (DNS).

In a second embodiment, the information collection module 2010 would extract data directly from a database. Consequently, a dialog box would not be necessary. However, it would be possible to provide some user interface to direct the network administrator as to how to extract the relevant information. Moreover, there could be informational screens allowing the network administrator to verify incorrect information prior to generating the executable file. Preferably, information collection module 2010 would identify particular fields of the database that are needed by the compiler module 2020 and extract only that information. Additionally, once the information is extracted, the information collection module 2010 would preferably display the information for the network administrator to verify. Upon his verification, the network administrator would preferably cause the information to be transferred to the compiler module 2020.

According to the second embodiment of the information collection module 2010, any database design can be used. Preferably, the database would include at least the same information described above with respect to the first embodiment. Specifically, the database would include the information necessary for the remote user to negotiate to the router/gateway and then onto the LAN. Additional detail concerning this information will be provided below.

In both the first and second embodiments of the information collection module 2010, information regarding the remote user's e-mail address and security code can also be provided. In this way, the compiler module, upon generating the executable file, could automatically transmit all generator programs (including the configuration program and a dialog program) to the remote user. Moreover, the executable file could be encrypted, using the security code as the decryption key. Alternatively, if the program will be provided to the remote user by a method other than e-mail, a storage address could be provided to the information collection module 2010, identifying where the executable files should be stored. For example, it could be stored in a directory on a computer or to a removable storage unit, such as a floppy disk or a compact disc.

Referring again to Figure 3, the compiler module 2020 is preferably software configured to generate an executable program using the data collected from the information collection module

2010. The executable program, when run on the remote computer, preferably configures the remote computer to be able to establish the VPN connection with the local network. In essence, the executable file is configured to install the VPN configuration onto the remote user's computer. It should be understood that the executable file could be used to configure their remote computer in
5 any fashion so as to customize it to a user's needs. For example, an executable program could be generated to allow a remote user to configure a program on the remote computer, such as Microsoft Outlook, to be set up in a certain way. Thus, if a user wanted both their work computer and their home computer to have identical settings for a particular program, a network administrator could generate an executable file to configure the home computer to have the same settings as the
10 were computer. The executable file would then be run on the home computer to change the configuration settings.

In the example of setting up the VPN connection, the compiler module 2020 preferably uses the information collected through the information collection module 2010 to generate an executable file that can be used by the remote user to configure the remote computer for the VPN
15 connection. In the preferred embodiment, the executable file is configured to edit the Remote Access Service (RAS) files of the remote computer's operating system. Specifically, the RAS phonebook is preferably manipulated. It should be understood that any method of modifying the registry entries of the remote computer's operating system could be used. This could include directly modifying .INI files and .INF files, or any other configuration files for the remote computer.

20 In the preferred embodiment, the data that is received from the information collection module 2010 is either embedded or attached to the executable file, such that upon execution of the executable file on the remote computer, the data is used to configure the remote computer. In the preferred embodiment, the Microsoft installer program is used as the executable file. Microsoft installer is a binary resource out of the OCX program module. Thus, the installer is the first
25 program built. The binary resource includes the configuration data, and because it is the first program to be built, the configuration data is initially empty. That is, the installer program is built with blank data.

When the information collection module 2010 generates the configuration data file, the blank data on the installer is replaced with the configuration data file. The installer program is
30 preferably configured to call the RAS API, and use the data configuration file to modify the RAS entries. It should be understood that the executable file to use any method to modify particular registry entries. Accordingly, the RAS entries for VPN that have been built are attached to the installer program, and the executable file for configuring the remote computer is created.

Additionally, the compiler module 2020 could be configured to automatically e-mail the executable file to an intended recipient. Thus, part of the information collected by the information collection module 2010 would include the e-mail address of the intended user. Once the code is written by the compiler module 2020, it would automatically be inserted into an e-mail addressed to the intended recipient. This e-mail could either automatically be transmitted when the executable file is generated, or could await a user command prior to sending the e-mail. Thus, for example, the configuration program generator could be integrated with any type of e-mail program, such as Microsoft Outlook or Microsoft Outlook Express.

When integrated with an e-mail program, the executable file is preferably written to an e-mail manager, such as an outbox or a draft box of the e-mail program. That is, it is written to a path on the computer that corresponds to the outbox. Moreover, an outlook plug-in will interpret specific data within the database as an e-mail address. Accordingly, the e-mail is automatically generated with the executable file or files as an attachment. Additionally, the body of the e-mail can be configured to include instructions for running the executable file. These instructions can be either tailored to a particular operating system of the user, or all instructions can be included, allowing the user to pick the appropriate set of instructions.

It should be understood that information collection module 2010 and the compiler module 2020 could be formatted as a single dialog box, so that once the information is entered into the dialog box, a single click to accept the data would generate the executable file. Additionally, that single click could also be configured to automatically send the e-mail to the intended user. Moreover, it should be understood that the whole process could be transparent to the network administrator, and thus selecting a remote user from the database would automatically generate the executable file, and could also automatically e-mail the executable file to the remote user. Alternatively, the process could be partially transparent. That is, the network administrator would have an opportunity to review data extracted from the database prior to generating the executable file. Moreover, the network administrator may be required to manually e-mail the program to the recipient.

It should be understood that the information collection module 2010 could be a stand-alone program, or could be a module within another application program. For example, information collection module could be an OCX module and part of a Microsoft Access database. Alternatively, the program could be written and compiled in any programming language, and configured to perform the same tasks. In the preferred embodiment, the information collection module is an OCX module.

Referring to Figure 5, additional details of the first embodiment (manual data entry) of the information collection module 2010 of Figure 3 are provided. As shown in Figure 5, a dialog box 200 of the configuration program generator according to a preferred embodiment is shown. The dialog box preferably includes a plurality of tabs to distinguish between information screens. In the 5 preferred embodiment, a WAN profile tab 205 and a user configuration tab 305 are provided. The WAN profile tab 205 corresponds to the network information collection module 3020 of Figure 4, and the user configuration tab 305 corresponds to the remote user information collection module 3010 of Figure 4. Figure 5 shows details of the dialog box with the WAN profile tab selected. Details of the user configuration tab 305 will be described below with reference to Figure 6.

10 The WAN profile tab 205 provides a user interface for the system administrator to enter information relating to the network to which the remote user will connect. First, a WAN text box 215 is provided. It is preferably configured in a format to accept a WAN number. The WAN number describes the IP address for the router/gateway. This information is used by the remote computer to identify the router/gateway to which it will connect to establish the VPN connection. This number 15 is preferably a static IP address that uniquely identifies the WAN to the outside world. Additionally, a gateway text box 230 is also provided. Within this text box, the administrator preferably enters the private gateway information. This address, also formatted as an IP address, cannot be seen by the outside world when they look at the router/gateway on the Internet. Again, this text box is preferably formatted as a standard IP address. Thus, the default dialog box is preset with all zeros, 20 such as "0.0.0.0". In the preferred embodiment, when a prescribed number of digits is entered prior to any one of the dots, the cursor automatically moves to the next digit in the address.

Next, a text entry box is provided for the primary and secondary DNS addresses 235, 240. The DNS (domain name system) is the way that Internet domain names are located and translated 25 into IP addresses. Various lists of domain names and IP addresses are distributed throughout the Internet in an hierarchical structure, rather than maintaining a central list of domain name-to-IP address translations.

Next, a text entry box is provided for the primary and secondary WINS addresses 255, 260. The WINS address points to location on the network where the Windows Internet Naming Service 30 (WINS) resides. WINS is an operating system based service that manages the association of workstation names and locations with IP addresses. This is done without a user's or an administrator's input. WINS thus automatically creates a mapping entry in a table to map a computer name to an associated IP address. Each computer name is unique. If a computer is moved to another geographic location, the subnet part of the IP address will most likely change. WINS ensures that the new subnet information will be updated automatically in the WINS table.

WINS works in conjunction with the Dynamic Host Configuration Protocol (DHCP), which negotiates an IP address for any computer when it is first defined to the network.

Figure 6 illustrates additional details of the first embodiment of the information collection module 2010 with the user configuration tab 305 selected. The user profile configuration box is preferably divided into two sections. The first section is the router/WAN section 5010, and the second section is the LAN section 5020. Within the LAN section 5020, there is a free formatted text box for the LAN IP address 5060. This text box is preferably presented with all "0" entered in each segment of the IP address. Additionally, the text box is preferably preformatted with dots between the initial zeros, for example having "0.0.0.0" as the initial address. Next, text boxes for login 5070 and password 5080 are presented. These are the login and password necessary for the user to access LAN. Finally, a drop-down box 5090 is provided to select the operating system of the remote computer. Any operating system could be selected from this drop-down box. Based on the selection of operating system, the executable file will be generated to configure particular aspects of the corresponding operating system. In the preferred embodiment, the LAN password is not needed. The LAN login is used to change the name of the computer. Thus the user will be prompted to give the password. Also the system can be configured to prompt for both the user name and password.

On the router/WAN side 5010, a text box is provided for each of login 5030 and password 5040. These represent the login/password combination necessary for the remote user to access the router. Additionally, there is a duplicate login check box 5050 provided to indicate whether the login/password combination for the router/WAN is the same as the login/password combination for the LAN. By checking this box 5050, the login/password combination entered on the router/WAN side 5010 is automatically entered on the LAN side 5020.

As with the WAN profile configuration dialog box, control buttons are provided for accepting or rejecting the information provided. These control buttons include an "OK" button 265, a "cancel" button 270, and an "apply" button 275.

According to the second embodiment, where a database is used to directly provide the information to the compiler module, the database would preferably have all the information described above with respect to Figures 40 and 50. Specifically, for each record of the database, fields for WAN information, WINS information, and DNS information would preferably be provided. Additionally, a user's login/password information would also be provided.

Referring to Figure 7, a construction of the database that is preferably used with the second embodiment is shown. It should be understood, however, that any method of storing and collecting data can be used, such as a spreadsheet, a text table, or comma separated values.

- Additionally, any layout of the database could be used. Preferably, the database will include modules for collecting and storing information needed to generate the executable program by the configuration program generator. For example, there may be a users module 6005, WAN settings 6010 module, ISP module 6015, and an operating system module 6020. The users module 6005
5 may require, for example, user identification information, user first name information, user last name information, user phone number information. Additionally, it may require user IP address information, user WAN login information, user WAN password information, user LAN login information, and user LAN password information. Additional user information that is not relevant to generating the executable file may also be stored in the database.
- 10 The WAN setting module 6010 is preferably configured to store information such as WAN settings identification information, company identification information, ISP identification information, hardware make information, hardware model information, hardware administrative password information, workgroup or domain, network type identification information, server type identification information, IP assignment type identification information, Gateway IP information, WAN IP
15 information and, primary DNS information, secondary DNS information, primary WINS information, secondary WINS information, and WAN setting notes. At a minimum, however, the WAN setting module 6010 should include the information required by the configuration program generator to generate the executable file.

Next, the operating systems module 6020 is preferably configured to store information such
20 as operating system identification information and operating system name. The ISP module 6015 is preferably configured to store information such as ISP identification information, ISP name information, ISP address information, ISP phone information, primary DNS IP information, and secondary DNS IP information.

Other information that may also be provided based on the particular need of the user may
25 include information such as contact information 6025, billing information 6030, employee information 6035, company information 6040, calls information 6045, user calls information 6050, ISP calls information 6055, contact types 6060, and any other pertinent information.

Additionally, the database is preferably a Customer Management Program (CMP). As such, it preferably includes full support and fields for troubleshooting the database. Additionally,
30 the database could include personal information regarding each user to validate the users. Such information could be mother's maiden name, place of birth, Social Security number, or any information that could authenticate a particular user.

Additionally, the database could be configured so that information common to all users would not have to be entered into the database each time a new user is added. For example, if all

users in the database will be establishing a VPN connection to the same network, information such as the DNS numbers and the WAN/Gateway address could automatically be inserted into the appropriate fields of the new record. Alternatively, if the database is used to manage a plurality of networks having different configuration settings, all users associated with any given network could
5 be identified (for example by using a company name) as belonging to the same group. When that group identification is provided to the database for a new user, the proper information for that user could automatically be populated into the fields of the database.

Moreover, other information to be automatically inserted into other fields for each record. For example, if it is determined that the client is Microsoft, when the Microsoft client is selected for
10 a new record in the database, the database will automatically insert all associated Microsoft features.

It should be understood that any database configuration or structure can be used, and the particular modules described above need not be present. Moreover, the database can be any type of the database, such as SQL or Microsoft Access. Any method of storing some or all of the data
15 necessary to generate the executable files by the configuration program generator would be suitable. Moreover, in the preferred embodiment, the information from the database is provided directly to the configuration program generator to generate the executable file for any given remote user. This eliminates the need for the network administrator to manually enter the information.

Additionally, when the database is used to generate the e-mail, string tables for the mailer
20 modules are provided. This information is interpreted as an e-mail address.

Next, a description of how a network administrator would manually use the above described system according to the first embodiment is provided. It should be understood that this process could be circumvented by associating the database with the configuration program generator, allowing the network administrator to simply select a user from the database and
25 automatically generate the executable file according to the data in the database, in accordance with the second embodiment.

Referring to Figure 8, using the manual procedure of the first embodiment, a network administrator would first preferably open a dialog box of the configuration program generator, as shown in step 200. This dialog box represents the information collection module. The dialog box
30 preferably includes a plurality of tabs to distinguish between various information screens. The network administrator would then select one of the tabs to determine which information screen would be displayed. Thus, the administrator could choose the WAN profile tab, as shown in step 205, or choose the user profile tab, as shown in step 210.

Depending upon the selected tab, the network administrator would then enter the appropriate information for the information screen that is displayed. Thus, if the administrator had chosen the WAN profile tab in step 205, the administrator would then proceed to enter WAN profile information, as shown in step 215. Alternatively, if the administrator had chosen the user profile tab 5 in step 210, then the administrator would proceed to enter user profile information, as shown in step 220.

In the preferred embodiment, both WAN profile information and user profile information need to be entered. Thus, upon entering information corresponding to one of the information screens, the administrator would next show the other information screen. When information for 10 both information screens have been entered, the administrator would apply the settings to generate the binary data file, as shown in step 225. The binary data file is then used to generate the executable file to be deployed to the remote user.

Additional details for each of the selectable dialog box tabs will be provided next. Figure 9 illustrates the steps to be followed by the network administrator if the network administrator selects 15 the WAN profile tab in step 205 of Figure 8, and Figure 10 illustrates the steps to be followed by the network administrator if the network administrator selects the user profile tab in step 210 of Figure 8. It should be understood that the steps need not necessarily be performed in the sequence shown. For instance, the network administrator can enter the information in any sequence, so long as all necessary information is provided. However, for sake of explanation, a particular sequence is 20 outlined below.

Referring to Figure 9, the network administrator selects the WAN profile tab in the information collection module dialog box. In the preferred embodiment, the dialog box is displayed on a computer monitor, and includes text boxes in which to input various information. The text boxes are labeled with a description of the information to be provided in the associated text box. 25 Additionally, for text boxes that require network addresses, the text boxes are preformatted as blank IP addresses, and with a field size limit for each number to be entered. Thus, it would be impossible to enter and incorrect number format for any given network address.

Next, as shown in step 310, the network administrator first enters a WAN profile number corresponding to the LAN/gateway/router. The WAN number preferably describes the IP address 30 for the router/gateway. This information is used by the remote computer to identify the router/gateway to which it will connect to establish the VPN connection. This number is preferably a static IP address that uniquely identifies the WAN to the outside world.

Next, as shown in step 320, the administrator would identify the type of network configuration of the LAN. For example, the network type could be Novell, NetWare, or Microsoft. It

should be understood that any kind of network environment could be used. For the purposes of example, it will be assumed that Microsoft is selected. By selecting the type, both the client and file/printer sharing our determined.

- Next, the administrator would preferably enter the type of the LAN, as shown in step 325.
- 5 For example, the LAN could be configured as a Server or as a Domain Controller. It should be understood that any LAN configuration can be implemented at this point. If the server type is selected as "Server," as shown in step 340, then the administrator next identifies the workgroup, as shown in step 345. The Workgroup information determines what group of users exchange access to resources on the network. The workgroup information is preferably maintained on the server.
- 10 Alternatively, if the administrator selects "Domain Controller" as a server type, as shown in step 330, and administrator would be prompted to fill in the domain information, as shown in step 335. In this scenario, the administrator would enter the Domain Controller group name needed to access the network. The domain information is used to uniquely identify a particular user.

The network administrator would next provide the gateway address information for the router/gateway, as shown in step 350. The gateway address is the IP address of the LAN. It may be public, and on the outside world, but may also be private and not known to the outside world. This IP address is for the network to see out from the LAN. Thus, when a user logs on to the network using the VPN connection, that user will access the Internet through the network, as opposed to through his own Internet connection.

20 Next, as shown in step 355, the administrator would provide the primary and secondary DNS addresses. The DNS (domain name system) is the way that Internet domain names are located and translated into IP addresses. Various lists of domain names and IP addresses are distributed throughout the Internet in an hierarchical structure, rather than maintaining a central list of domain name-to-IP address translations. In the preferred embodiment the administrator would determine the DNS number by querying the ISP. However, an administrator may receive DNS information from any other source. The administrator would also preferably enter a second DNS number as received for the ISP. However, only one DNS number is necessary for the operation.

30 Next, as shown in step 360 the administrator preferably enters at least one WINS number, and optionally enters a second WINS number as well. The WINS address points to a network location where the Windows Internet Naming Service (WINS) resides. This is typically located on the server, although it could be located on any machine on the network. Additionally, the WINS information is preferably known by the administrator based on the network configuration, although this information can be obtained by the administrator from any other source.

WINS is an operating system based service that manages the association of workstation names and locations with IP addresses. This is done without a user's or an administrator's input. WINS thus automatically creates a mapping entry in a table to map a computer name to an associated IP address. Each computer name on the network is unique. If a computer on the 5 network is moved to another geographic location, the subnet part of the IP address will most likely change. WINS ensures that the new subnet information will be updated automatically in the WINS table. WINS typically works in conjunction with the Dynamic Host Configuration Protocol (DHCP), which negotiates an IP address for any computer when it is first defined to the network.

The administrator next optionally selects an IP assignment from a plurality of IP 10 assignments, as shown in Step 370. The IP assignment to be one of public and private, as shown in steps 371 and 372. Moreover, as shown in steps 375a ~375c, the administrator may select a router option, a server option, and a static IP option.

The IP addresses the remote computer's IP address on the network. The assignment of 15 the IP address can be set by the DHCP router or the DHCP server, or a static IP address can be given by the system administrator. If all zeros are initially used, this typically signals that the DHCP should be prompted to assign an address.

It should be understood that the WAN profile dialog box could prompt the network 20 administrator for additional information. However, in the preferred embodiment, the above information is sufficient. Thus, if the user information has already been provided (under the "user" tab of this configuration dialog box), the network administrator would proceed to generate the user data binary file. However, if the user data information has not yet been provided, the network administrator would select the user information tab to display the user information portion of the information collection module dialog box.

Figure 10 illustrates a method of providing remote user information to the information 25 collection module according to the first embodiment. Thus, referring to Figure 10, the network administrator selects the user profile tab in a dialog box. Once that is selected, a user interface module dialog box appears, which prompts the network administrator to enter user specific information. This information is used to enable the remote user to login to the router/WAN, as well 30 as to login to the LAN. In the preferred embodiment, the information gathered from this dialog box is used to generate the automatic login portion of the executable file. It should be understood that, except for the remote computer's operating system, this information is not necessary to establish the VPN connection; rather, it is used to provide a method of automating the login process to make it easier for the remote user. Thus, if this information is provided, after the remote computer has been configured by running the configuration executable file, a "Dialer" program appears as an

executable file on the remote computer. The remote user can then simply select the dialer program to be automatically connected to the LAN.

In the absence of this information, after the computer is configured for the VPN connection, the user would be faced with several login/password dialog boxes before gaining access to the
5 LAN. For some users, this becomes cumbersome. Consequently, in the preferred embodiment, the network administrator provides the login/password information for the user to the configuration program generator prior to generating the executable file. This information could be provided manually or could be extracted from the database. By providing the login/password information, the configuration program generator is adapted to generate a Dialer program that could be
10 deployed on the remote computer. When the remote user executes the Dialer program on the remote computer, and if the remote computer has been properly configured for the VPN connection, a VPN connection between the remote computer and the local network is automatically established.

It should be understood that the Dialer program could be generated and deployed to the
15 remote computer separately from the network settings executable file. For example, the dialer program could be sent to the remote computer by a separate e-mail, or any other file delivery method. Consequently, should the login/password combinations for the remote user change at any time, the network administrator could provide a new dialer program having the revised login/password information. Additional information regarding the Dialer is provided hereinbelow.

20 Referring again to Figure 10, the network administrator first enters the LAN IP address, as shown in step 410. This information preferably corresponds to the LAN that the remote user will access. Next, as shown in step 420, the administrator enters the remote user's login name and password for the router/WAN. Additionally, the network administrator enters the remote user's login name and password for the LAN, as shown in step 430. It should be understood that if this
25 login/password combination is identical to the router/WAN login/password, the administrator can simply select a check box indicating "duplicate login." By doing this, the login/password information for the LAN is automatically copied from the router/WAN information.

Finally, in step 440, the administrator identifies the operating system that the remote computer is using. This is necessary for the executable configuration program to properly modify
30 the operating system parameters to establish the VPN connection. In the preferred embodiment, Windows XP, Windows 2000, Windows ME, or Windows 98 appear in a drop-down menu box. However, it should be understood that any operating system could be selected, including UNIX or LINUX, or any successor to the Windows family of operating systems. Moreover, and the preferred embodiment, a single executable files will work with all supported offering systems.

If the network information has already been provided (under the "WAN profile" tab of this configuration dialog box), the network administrator would proceed to generate the user data binary file. However, if the network information has not yet been provided, the network administrator would select the "WAN profile" tab to display the network information portion of the information collection module dialog box.

The user data binary file is then used when generating the executable file. This information is interpreted by the remote computer so as to configure the remote computer for establishing a secure Internet connection to the LAN. The secure Internet connection between the remote computer and the LAN is the Virtual Private Network (VPN). In a preferred embodiment, Point-to-Point Tunneling Protocol (PPTP) is used for establishing the VPN connection. PPTP is a Layer 2 tunneling protocol which allows the remote computer to use a public IP network to communicate securely with the local computer (for example, the LAN server). PPTP thus essentially tunnels the IP. A VPN "tunnel" between the LAN and the remote computer allows traffic to flow securely across the Internet. The PPTP tunnel is preferably established using the Dial-Up Networking functions of the Windows operating system. Remote users are preferably authenticated using MS-CHAP against a user/password list maintained. Other authentication schemes, however, could be used. PPTP is described in RFC 2637, "Point-to-Point Tunneling Protocol (PPTP)," which is incorporated herein by reference. Consequently, in the preferred embodiment, the Dial-Up Networking aspects of Windows are manipulated by the executable file. For example, the RAS phonebook entries could be modified or registry entries pertaining to Dial-Up Networking could be modified.

Alternatively, IPSec could be used instead of PPTP. IPSec includes a series of protocols to handle data integrity and confidentiality when transmitting secure data across the Internet. A first protocol, the Authentication Header (AH) protocol handles data integrity, and a second protocol, the Encapsulated Security Payload (ESP) protocol, resolves both data integrity and confidentiality issues.

Each VPN tunnel created with IPSec is preferably assigned a specific Security Parameter Index (SPI) to differentiate it from other VPN tunnels configured on router/gateway. An SPI is an arbitrary, 32-bit number that tells the receiving device which group of security protocols (such as algorithms, keys and their longevity) the remote user is using for communication.

IPSec is described in RFC 2401, "Security Architecture for the Internet Protocol," and RFC 2709, "Security Model with Tunnel-mode IPsec for NAT Domains," which are incorporated herein by reference. The same general theory would apply to IPsec as applies to PPTP with respect to configuring the remote computer to establish the VPN connection.

Additionally, it should be understood that the configuration method described herein can be applied to any communication protocol. For example, in addition to PPTP and IPsec, L2TP could also be used, or any other protocol, especially any other tunneling protocol.

Additional information concerning the Dialer program is next provided.

5 The Dialer program is a application that functionally performs dialing functions using any operating system. According to the preferred embodiment, its calls dialing functions contained with Microsoft "Dial-up Networking" contained within Microsoft Windows versions 95/98/ME and "Network and Dial-up Connections" within Microsoft Windows versions 2000/XP. The Dialer preferably assist in the operation of the connection method described herein by providing
10 navigation for the user, and it binds user announcements when activating a connection. For example, it provides a user with the ability to activate a VPN connection using a single "click" of a mouse.

15 Additionally, the dialer can assist in providing various security parameters within a remote access solution. For example, when making a connection "active" a user must provide information that validates that user for security features. This provides protection of both the "Caller" (the remote user making a secure connection) and the "Callee" (the network topology with this remote access feature). Every secure VPN or network connection requires some form of verification. The Dialer provides a module for both user verification, as well as connection activation.

20 The Dialer preferably makes user navigation easier. When the user initiates and starts a connection to get to a destination network, at least one, and typically several connections need to occur to access the network.

25 For example, a user may first need to access the Internet. After the Internet connection is established, the user or the VPN might possibly need to initiate a connection "directly" to the device that holds the parameters of how that user is going to "talk" (i.e. communicate) to the network topology. This includes encapsulation and authentication, i.e. the WAN Profile of the Callee. The third connection is necessary to access the network structure, for example to join the "Network Neighborhood," from the remote location. This is performed by the internal validation (user accounts and privileges) within the network topology, and preferably occurs within a network server
30 or appliance when a new connection is established or sensed by the network server or appliance.

Pending on the number of network servers or appliances and security parameters within a network topology, each server can require a different validation of the same server. The Dialer eliminates the second of the stated connections because when the user has some or any

connection to the Internet, the Dialer brings the "Caller" through the Internet to the "Callee" using a single "click."

- To perform these functions, the Dialer preferably contains all the required announcements and authentications the user needs to both start and complete a secure connection to the network.
- 5 Thus, the dialer contains programming code configured to select and initiate a specific dialing function. For example, the dialer can be configured to select and initiate a specific "RAS phonebook entry" that has been created using the above described configuration process.

- Additionally, the Dialer replaces the "default" announcement program that is bound to any connection profile built in the operating system's dialing functions. For example the dialer would
- 10 replace the connection profile contained in the "Dial-up Networking" contained within Microsoft Windows versions 95/98/ME, and would replace the "Network and Dial-up Connections" within Microsoft Windows versions 2000/XP.

- By using the Dialer, user does not need to know the user's particular login and password combination to access the WAN of the network. Additionally, the Dialer will not work on any RAS
- 15 phonebook entry that is not a match to the corresponding label. In doing this the dialer can direct both security and how the user connects because without the Dialer no connection can or would be made.

- In the preferred embodiment, the Dialer is built from the user settings index within the database. Additionally, it is specific to the WAN Login and WAN password for a particular user. At
- 20 both input fields the remote access engineer can configure and design a custom Dialer that is very complex containing a series of numbers, letters and characters, completely random" and unknown to the user. However, because this information is known to the dialer, access will be granted to the network.

- Thus, when the remote computer executable configuration file is generated to establish the
- 25 VPN connection, the network administrator preferably deploys the executable file to the remote user. This can be done by any method of transferring data, including attaching the program to an e-mail or providing on a computer readable medium. Additionally, if a Dialer executable program has been generated, it can be deployed simultaneously with the configuration executable file. It should be understood, however, that each of these executable files can be deployed separate and
- 30 apart from the other executable file. For example, if a user is required to change a login/password combination according to a prescribed time schedule, the network administrator could deploy the dialer executable program to the remote user with the updated login/password combination at the appropriate time. When the remote user executes this updated program, the login/password settings on the remote computer are updated. Additionally, according to the preferred embodiment,

the executable configuration file is self deleting. That is, upon execution on the remote computer, after the remote computer is fully configured the executable file is automatically deleted from the remote computer. Alternatively, the dialer program could be set up to search for the configuration program the first time the dialer program is executed. If the configuration program is found, then 5 the dialer program could be configured to automatically delete the configuration program.

Moreover, and in order to enhance security, the configuration and dialer programs to be deployed to the user can be stored within a removable USB key-type memory device that is provided to the user. Referring to Figure 11, the USB key-type memory device 8010 is a small portable single piece device that plugs into a USB port of a computer using the USB interface connector 10 8020. The USB key-type memory device 8010 preferably includes an electronically erasable read-only memory (EEPROM) 8030. The USB key also preferably includes a flash RAM 8040. The EEPROM 8030 is preferably a locked EEPROM section, which prevents users from accessing information stored therein. The flash RAM 8040 is a general storage area, which emulates a disk/hard drive. The EEPROM 8030 can be write protected all the flash RAM 8040 is for storing 15 general data. The USB key-type memory device 8010 further includes a USB controller 8050 to control the device.

Because the EEPROM 8030 is inaccessible to users, it can be used to store verification and/or encryption information. According to the preferred embodiment, before the configuration program or the dialer program can be run, whether from a hard drive, a remote disk, or the flash 20 RAM 8040, the program would first query the EEPROM to authenticate the user. Thus, if the USB key-type memory device 8010 is not coupled to the USB port of the computer, neither the configuration program or the dialer program will work.

Thus, the dialer program and configuration program could initially be stored to the file structure section of the removable key. These programs could be run directly from the key, or 25 could be copied to a hard drive and run from there. However, without the USB key being inserted into the USB drive, the programs would not operate.

Specifically, as part of the configuration program and the dialer program, a special code could be incorporated that requires the program to access the EEPROM 8030 of the USB key 8010. Within that EEPROM 8030, a password could be stored that would authenticate the dialer 30 program and/or the configuration program.

To generate the USB key 8010, the administrator would first access the EEPROM 8030 of the USB key 8010 to program a password that would be used by another program. Accordingly, it would not be necessary for the remote user to even know the password; rather the programs to be executed in automatically query for the password. The executable program, in order to run, would

need to verify the existence and authenticity of the password on the EEPROM 8030 of the USB key 8010.

The executable programs, for example the dialer program or the configuration program, can be stored in the flash RAM 8040 of the USB key 8010. When the dialer program is run, it
5 would first query the EEPROM 8030 to determine if the password stored in the EEPROM 8030 is identical to that which is required by the dialer program. If the password matches, than the program will run. Otherwise, no connection can be made to the VPN. Even if the dialer program is copied to the hard drive, a would still require the authentications through the USB key. Thus, if the dialer program is copied to a hard drive other than the USB key 8010, it will not operate unless the
10 USB key 8010 is installed on the system.

Moreover, and order to protect the key for being stolen, a further layer of security could be used that requires the user to enter a password to authenticate the user prior to authenticating the key. Moreover, the dialer program could be encrypted, when the decryption key being stored in the EEPROM 8030 of the USB key 8010. Thus, would be very difficult, if not impossible, for someone
15 to modify the dialer program to run without use of the USB key 8010. It should be understood that the configuration generator program can also be so encrypted. Additionally, the information on the EEPROM 8030 could also be encrypted.

Additionally, because the installer program is intended to be run once while the dialer program is intended to be run every time a VPN connection is desired, each of the two programs
20 could have a unique password stored on the EEPROM 8030. When the installer program is run, it would call for the associated password stored on the EEPROM 8030. After installation, it would preferably erase that password. Consequently any further attempt to run the installer program would fail for lack of authentication. Meanwhile, the dialer program would have a different password associated with it on the EEPROM 8030. Accordingly, when the dialer program is run it would first
25 query the associated password on the EEPROM to determine the authenticity of the key. If the password were found, then the dialer program would proceed to establish a connection to the server.

The system and method for automatically configuring a remote computer according to the preferred embodiment has various advantages. For example, it allows a network administrator to
30 easily configure remote computers by generating a single executable file that can be provided to the remote user for execution. By doing this, the network administrator can save much time and can more actively manage the network. Additionally, any portion of a remote user's computer can be remotely configured by using an application program designed to manipulate the registry of the remote computer. Consequently, anyone with the need to remotely modify the configuration

settings of a computer can generate an executable file to be run on the remote computer to accomplish the configuration update. Additionally, by having a conduit from a database to the program modules, it becomes easy to generate a large number of such executable files in a short period of time. This allows for efficient management of remote users.

5 Moreover, according to the preferred embodiment, the remote user's computer is seen as a gateway in establishing the VPN connection to the host/local computer. Accordingly, because a VPN connection is typically established between gateways, it is possible to automatically establishing VPN connection between the local computer's gateway and the remote computer directly. Thus the remote computer essentially becomes a router.

10 The foregoing embodiments and advantages are merely exemplary and are not to be construed as limiting the present invention. The present teaching can be readily applied to other types of apparatuses. The description of the present invention is intended to be illustrative, and not to limit the scope of the claims. Many alternatives, modifications, and variations will be apparent to those skilled in the art. In the claims, means-plus-function clauses are intended to cover the 15 structures described herein as performing the recited function and not only structural equivalents but also equivalent structures.

WHAT IS CLAIMED IS:

1. A method of configuring a remote computer to access a network of computers, comprising:

5 initiating an installer program having an empty binary file, the installer program being configured to generate an executable installation program using the binary file;

generating a user data binary file including data necessary to configure the remote computer to access the network;

10 replacing the empty binary file with the user data binary file and generating the executable installation program using the user data binary file; and

executing the executable installation program on the remote computer to modify configuration settings of the remote computer such that the remote computer is configured to access the network.

2. The method of claim 1, wherein the executable installation program is e-mailed to
15 the remote computer to be executed on the remote computer.

3. The method of claim 1, further comprising storing the executable file on a computer readable medium prior to execution of the executable file to provide the executable file to the remote computer for execution.

20

4. The method of claim 3, wherein the computer readable medium comprises a universal serial bus (USB) data storage peripheral having first and second storage areas, the first area being an electronically erasable read-only memory (EEPROM) storage area and the second storage area being a flash RAM storage area.

25

5. The method of claim 4, wherein the first storage area is a secure storage area which is accessible to a user of the USB data storage peripheral, and wherein the second storage area comprises a general storage area, which emulates a disk/hard drive, wherein the executable file is stored in the second storage area.

30

6. The method of claim 5, wherein an encryption key is stored in the first storage area, and wherein the executable file searches the first storage area for a prescribed authentication key before it can be executed by a user.

7. The method of claim 6, wherein the modified configuration settings of the remote computer comprise a setting that requires the remote computer to access the first storage area of the USB data storage peripheral prior to accessing the network.

5 8. The method of claim 6, wherein the remote computer cannot access the network if the encryption key within the first storage area cannot be accessed by the remote computer.

9. The method of claim 1, wherein the user data binary file comprises an IP address of a router coupled to the network of computers, a gateway address of the router, at least one DNS 10 address, and at least one WINS address.

10. The method of claim 9, wherein the user data binary file further comprises an identification of an operating system used by the remote computer and an identification of an operating system of the network of computers.

15 11. The method of claim 10, wherein the user data binary file further comprises at least one login and password necessary for the remote computer to access the network of computers.

12. The method of claim 1, wherein the user data binary file is generated by an user 20 data application that receives network information needed by the remote computer to gain access to the network.

13. The method of claim 12, wherein the user data application prompts an operator of the network for the network information, and generates the user data binary file upon a command of 25 the operator.

14. The method of claim 12, wherein the user data application extracts the network information from a database containing relevant configuration information for the remote computer.

30 15. The method of claim 12, wherein the user data application is one of an OCX and ActiveX application.

16. The method of claim 15, wherein the user data application is configured to run with at least one of a commercial database application, a commercial spreadsheet application, and a commercial word processor application.

5 17. The method of claim 15, wherein the prescribed RAS settings configure the remote computer to establish a VPN connection with the network.

10 18. The method of claim 1, wherein the executable installation program installs prescribed Remote Access Service (RAS) settings on the remote computer, and wherein the prescribed RAS settings are adapted to allow the remote computer to access the network.

19. A method of configuring a remote computer to access a network of computers, comprising:

15 accessing a database having networking information necessary for at least one remote computer to access the network of computers, including a WAN profile for the network of computers, at least one DNS address for the network of computers, and at least one WINS address for the network of computers;

20 executing a software module from within the database, the software module configured extract the networking information for a selected one of the at least one remote computer;

converting the extracted networking information to a binary code representing configuration settings for the selected one of the at least one remote computer;

25 generating an executable program having the binary code embedded therein, the executable program being configured to modify a current configuration of the selected one of the at least one remote computer when executed by the remote computer; and

providing the executable program to the selected one of the at least one remote computer for execution.

20. The method of claim 19, wherein providing the executable program comprises at least one of e-mailing the executable program to the remote computer, transferring the executable program to a magnetic storage device, transferring the executable program to an optical disc, and loading the executable program onto a web site for download by the remote computer.

21. The method of claim 20, wherein the magnetic storage device comprises a universal serial bus (USB) data storage peripheral having first and second storage areas, the first area being an electronically erasable read-only memory (EEPROM) storage area and the second storage area being a flash RAM storage area.

5

22. The method of claim 21, wherein the first storage area is a secure storage area which is accessible to a user of the USB data storage peripheral, and wherein the second storage area comprises a general storage area, which emulates a disk/hard drive, wherein the executable file is stored in the second storage area.

10

23. The method of claim 22, wherein an encryption key is stored in the first storage area, and wherein the executable file searches the first storage area for a prescribed authentication key before it can be executed by a user.

15

24. The method of claim 23, wherein the modified configuration settings of the remote computer comprise a setting that requires the remote computer to access the first storage area of the USB data storage peripheral prior to accessing the network.

20

25. The method of claim 22, wherein the remote computer cannot access the network if the encryption key within the first storage area cannot be accessed by the remote computer.

26. The method of claim 19, wherein the networking information further comprises a login and password combination necessary for the selected one of the at least one remote computer to access the network of computers and an IP address for the network of computers.

25

27. The method of claim 26, wherein the networking information further comprises a login and password combination necessary for the selected one of the at least one remote computer to access a router/gateway that couples the network of computers to the Internet.

30

28. The method of claim 27, wherein the networking information further comprises an identification of an operating system used by the selected one of the at least one remote computer.

29. The method of claim 19, wherein the executable file is configured to modify a registry of the selected one of the at least one remote computer to insert the information necessary for the remote computer to access the network of computers.

5 30. The method of claim 29, wherein the executable file is configured to modify RAS phonebook entries of the remote computer for establishing a virtual private network (VPN) with the network of computers.

10 31. The method of claim 29, wherein the executable file is configured to modify dial-up networking settings of an operating system of the remote computer necessary to establish a secure Internet connection between the remote computer and the network of computers.

15 32. A method of configuring a remote computer, comprising:
accessing a database having configuration information necessary to configure at least one remote computer according to a user's preferences;
executing a software module associated with the database, the software module configured extract the configuration information from the database for a selected one of the at least one remote computer;
converting the extracted networking information to a binary code representing configuration settings for the selected one of the at least one remote computer;
generating an executable program having the binary code embedded therein that will modify a current configuration of the selected one of the at least one remote computer when executed by the remote computer; and
providing the executable program to the selected one of the at least one remote computer for execution.

30 33. The method of claim 32, wherein providing the executable program to the selected one of the at least one remote computer comprises e-mailing the executable program to the remote computer, transferring the executable program to a computer readable magnetic medium, transferring the executable program to a computer readable optical medium, and loading the executable program onto a web site for download by the remote controller.

34. The method of claim 32, wherein the configuration information comprises an identification of an operating system of the selected one of the at least one remote computer.

35. The method of claim 32, wherein the configuration information comprises application program settings, and wherein the executable program is configured to modify a registry of the remote computer.

5

36. A method of modifying configuration settings of a remote computer; comprising:
determining configuration settings necessary for a remote computer to perform a prescribed operation;

generating a binary file representing the configuration settings for the remote
10 computer;

generating an executable program having the binary code embedded therein that
will modify a current configuration of the remote computer when executed by the remote computer;
and

deploying the executable program to the remote computer for execution.

15

37. The method of claim 36, wherein the executable file is configured to modify a registry settings of the remote computer to effect the configuration modification.

38. A method of configuring a remote computer to access a network of computers,
20 comprising:

executing a software module that is configured to prompt a system administrator
for configuration settings of the remote computer to access the network of computers;

receiving the configuration setting provided by the system administrator through
the software module;

25 converting the configuration settings received through the software module to a
binary code representing the configuration settings;

generating an executable program that will modify a current configuration of the
remote computer when executed by the remote computer, the executable program having the
binary code embedded therein; and

30 providing the executable program to the remote computer for execution.

39. A computer readable medium having stored thereon a sequence of instructions which, when executed by a processor, cause the processor to perform a sequence of steps, comprising:

- 5 initiating an installer application having an empty binary file, the installer application being configured to generate an executable installation program using the binary file;
- generating a user data binary file including data necessary to configure the remote computer to access the network;
- replacing the empty binary file with the user data binary file and generating the executable installation program using the user data binary file; and
- 10 storing the executable installation program on a storage medium to be run by the remote computer to modify configuration settings of the remote computer such that the remote computer is configured to access the network.

40. A method of configuring a remote computer to establish a secure connection to a local computer, comprising:

- 15 determining configuration settings required by a remote computer to access a local computer;
- converting the configuration settings required by the remote computer to a binary code representing the configuration settings; and
- 20 generating an executable program that will modify a current configuration of the remote computer when executed by the remote computer, the executable program having the binary code embedded therein.

41. The method of claim 40, further comprising providing the executable program to the remote computer for execution.

42. The method of claim 40, wherein the configuration settings are determined by a network system operator and manually keyed into a conversion application, the conversion application being configured to convert the configuration settings to the binary code representing the configuration settings.

43. The method of claim 42, wherein the conversion application comprises a software module configured to be integrated with at least one of a database application software and a spreadsheet application software.

44. The method of claim 42, wherein the executable program is further configured to establish an auto-dialer feature that automatically establishes a connection with the ISP and automatically provides a user name and password to provide an automatic login to the network.

5

45. The method of claim 40, wherein the current configuration to be modified by the executable program comprises at least one of RAS entries, a windows registry and Windows Phone Book files.

10

46. The method of claim 40, wherein the executable program is automatically deleted from the remote computer upon its execution by the remote computer.

15

47. The method of claim 40, wherein the configuration settings required by the remote computer comprise at least one DNS number, at least one WINS address, and a WAN profile of a gateway coupled to the local computer.

48. The method of claim 40, wherein the local computer comprises a gateway/router and a local area network (LAN).

20

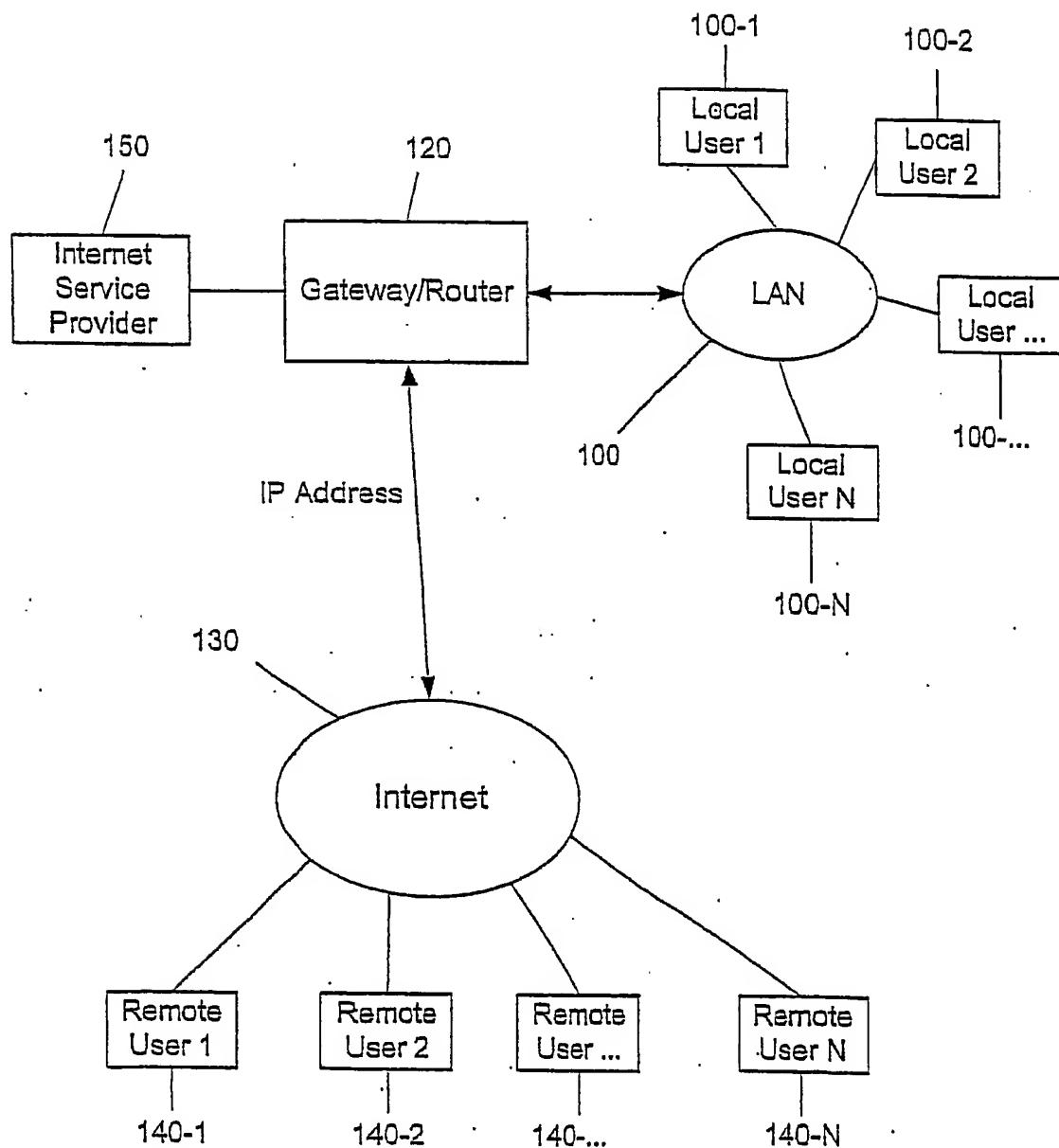
Figure 1

Figure 2

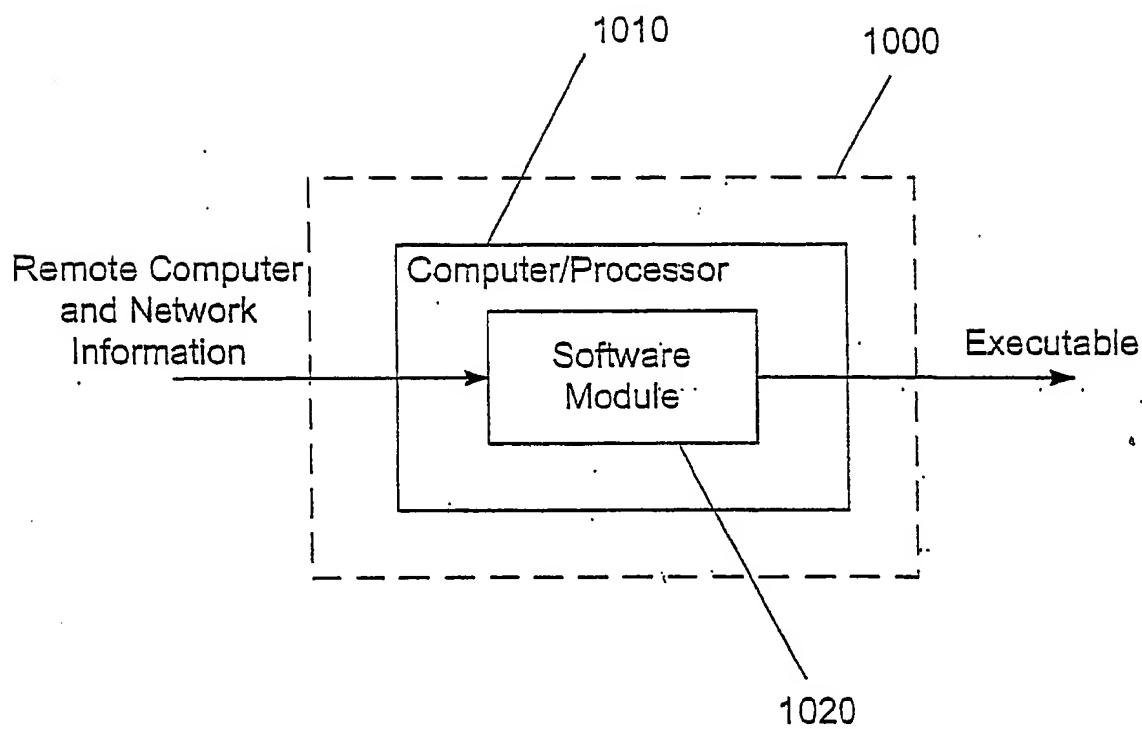


Figure 3

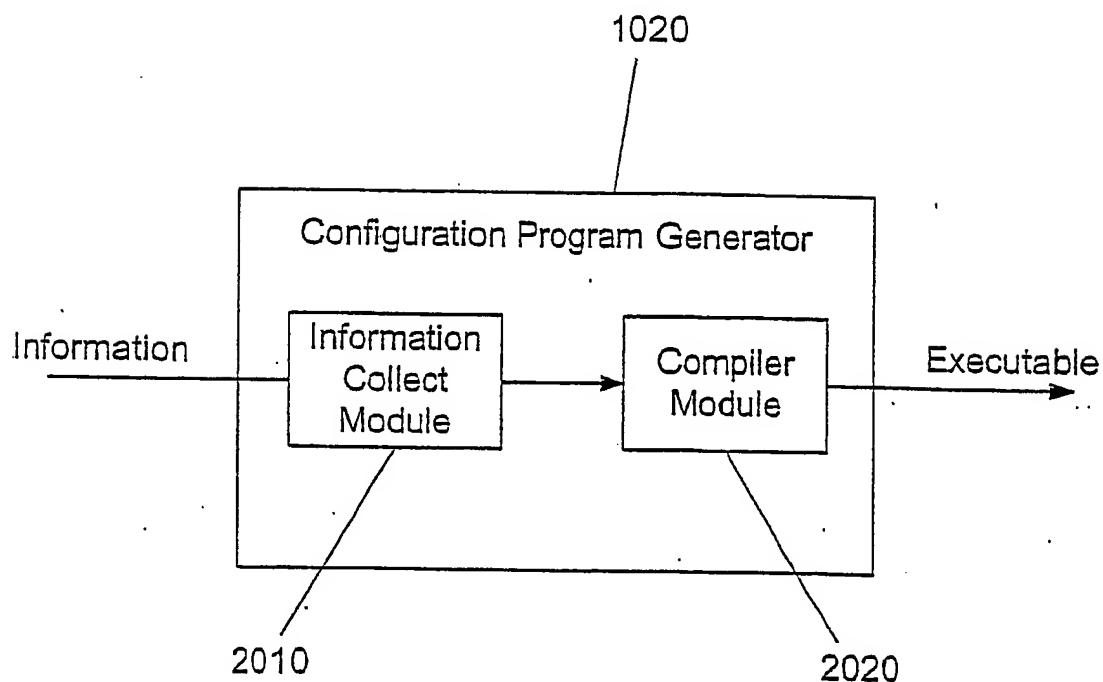
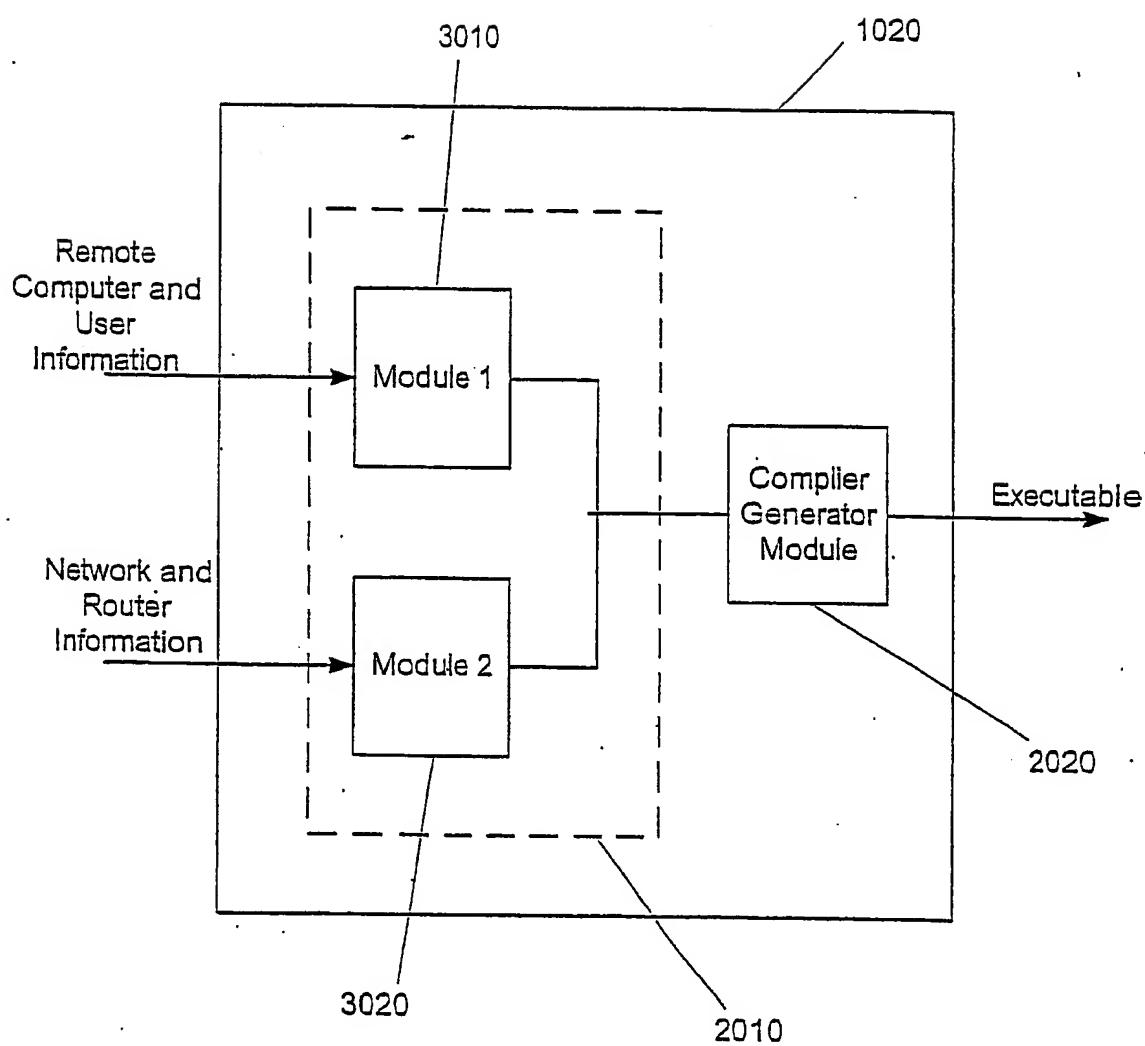


Figure 4



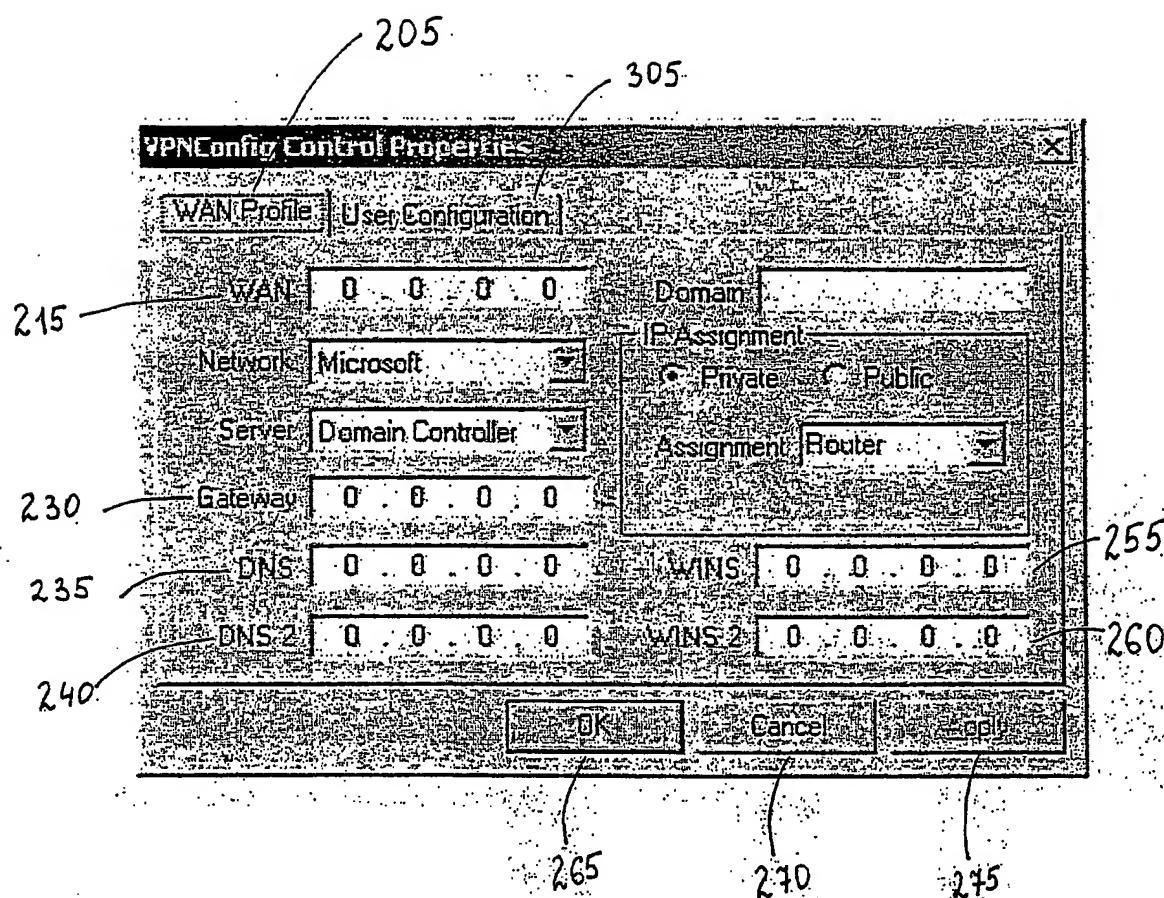


Figure 5

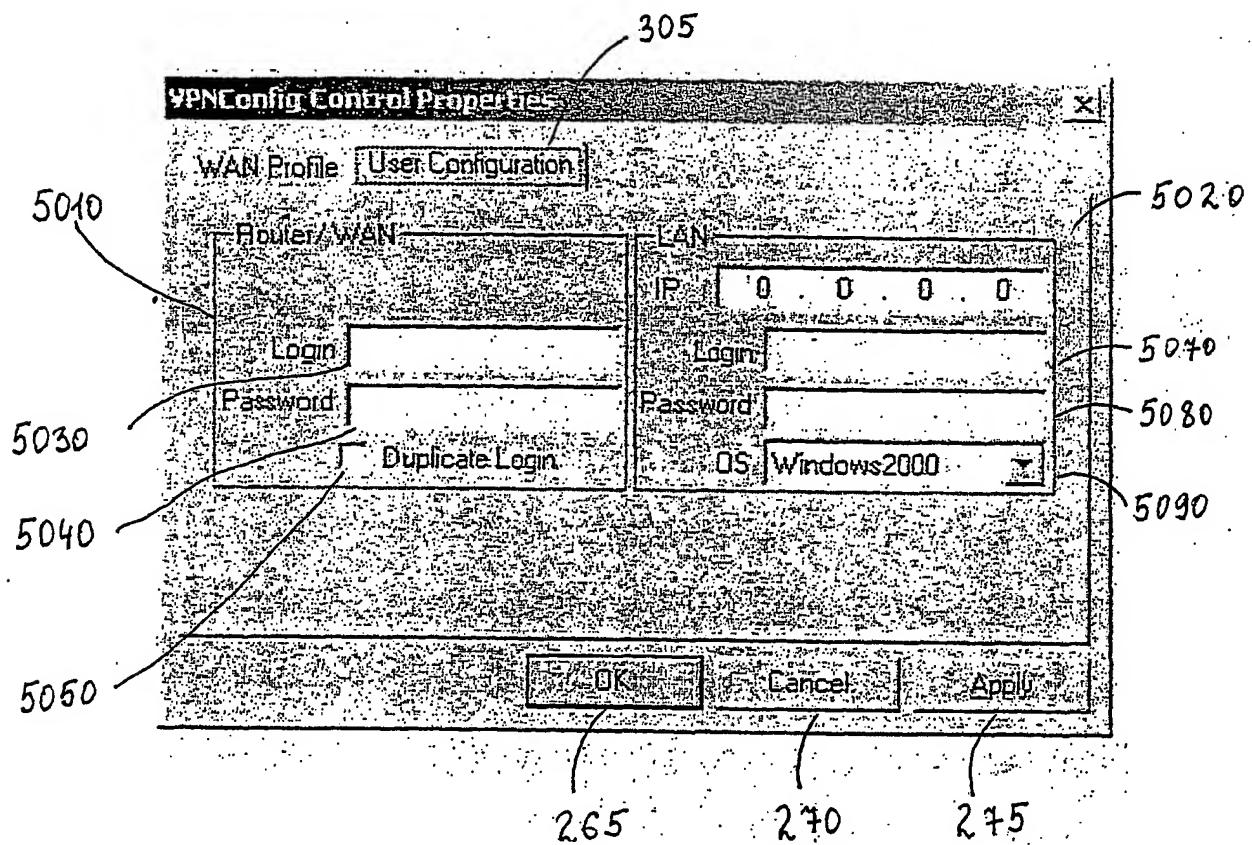


Figure 6

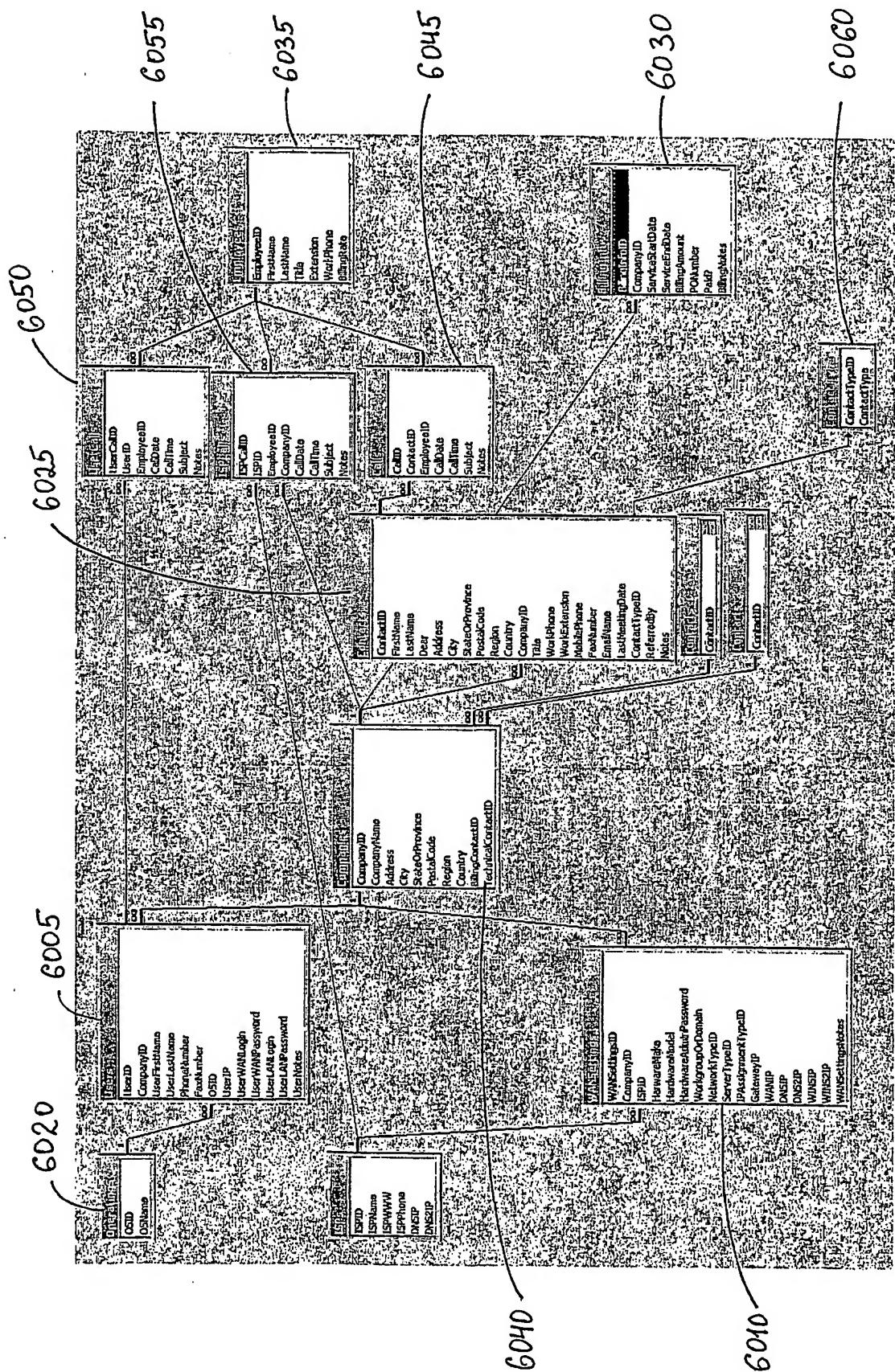


Figure 4

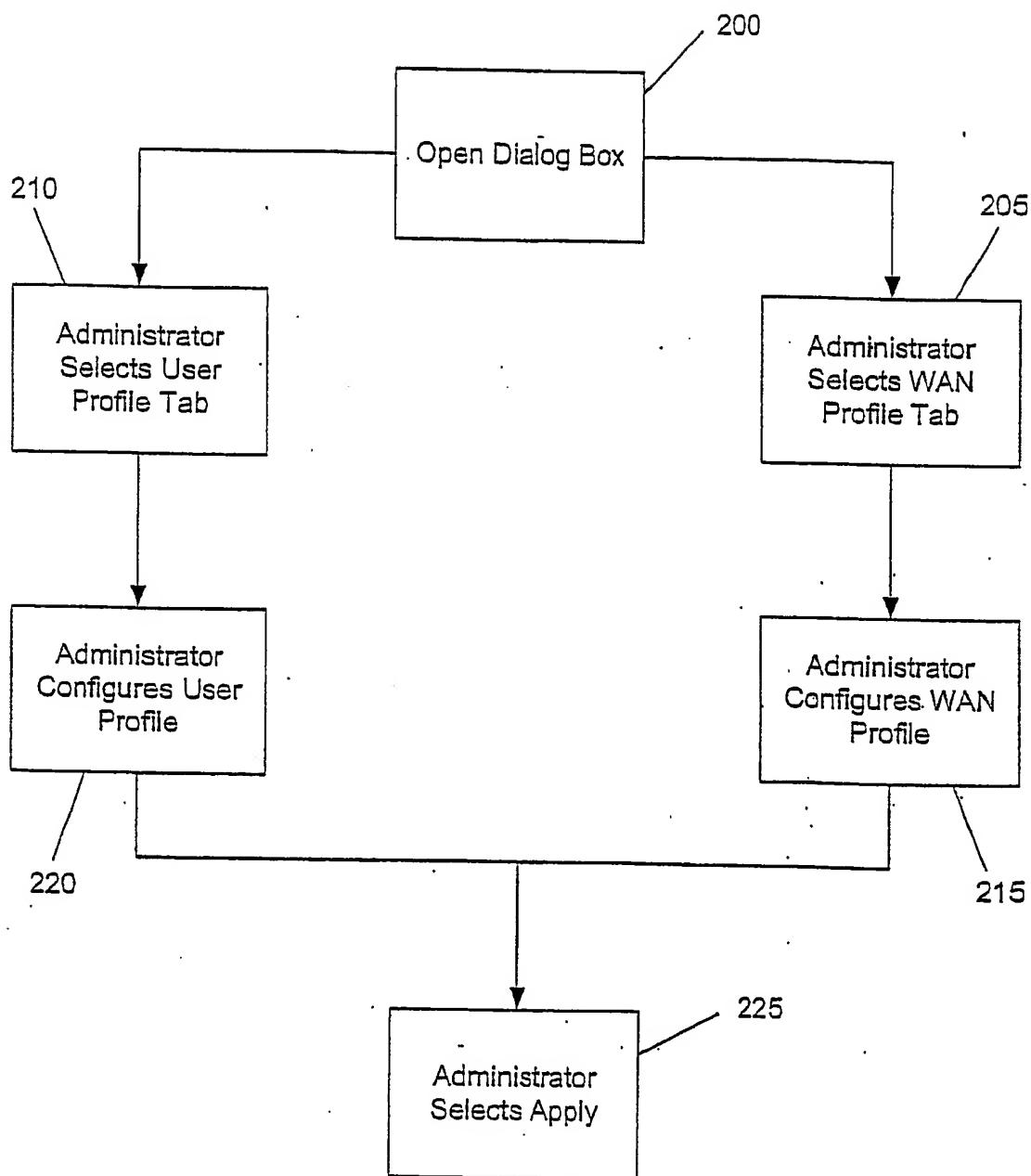
Figure 8

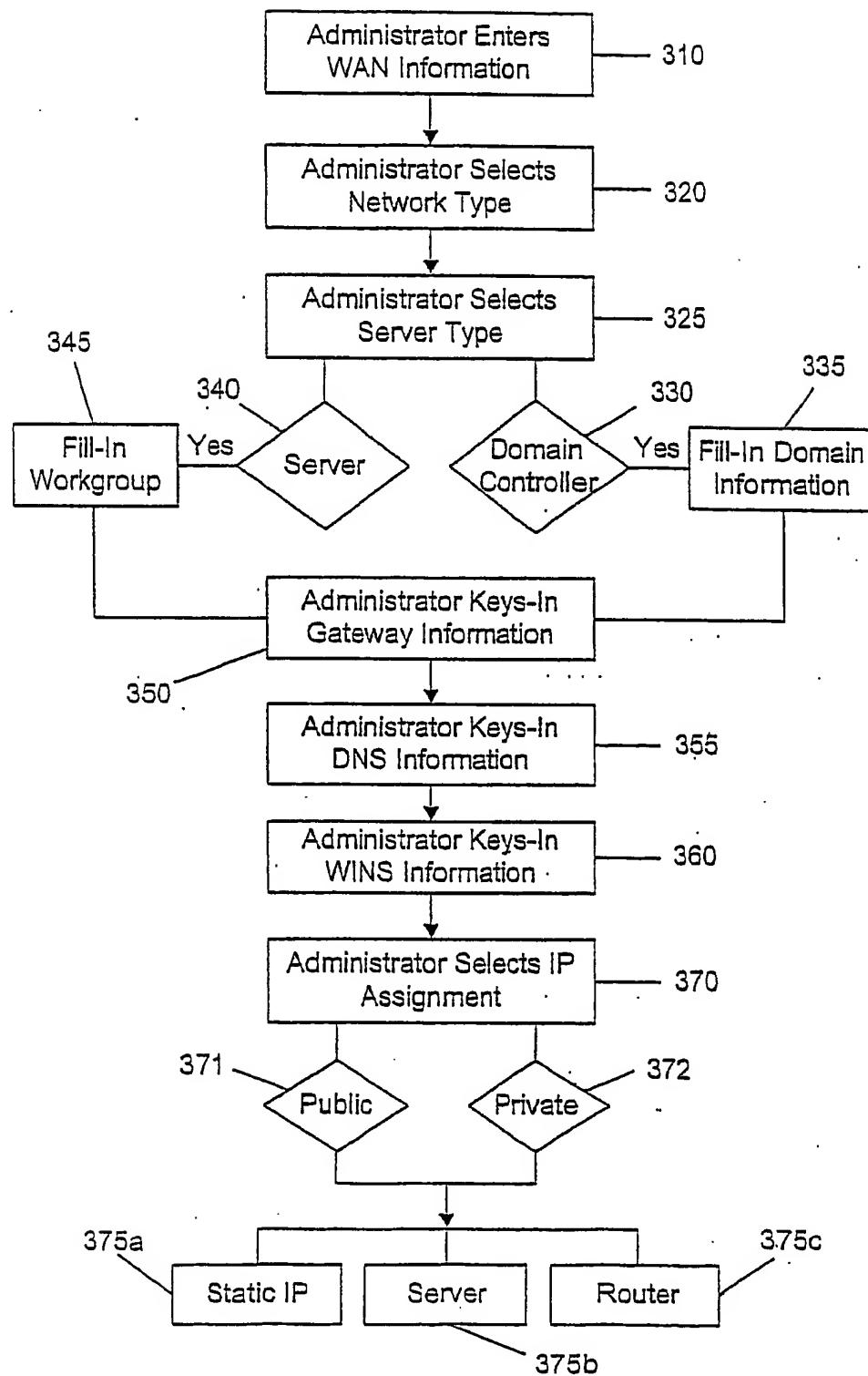
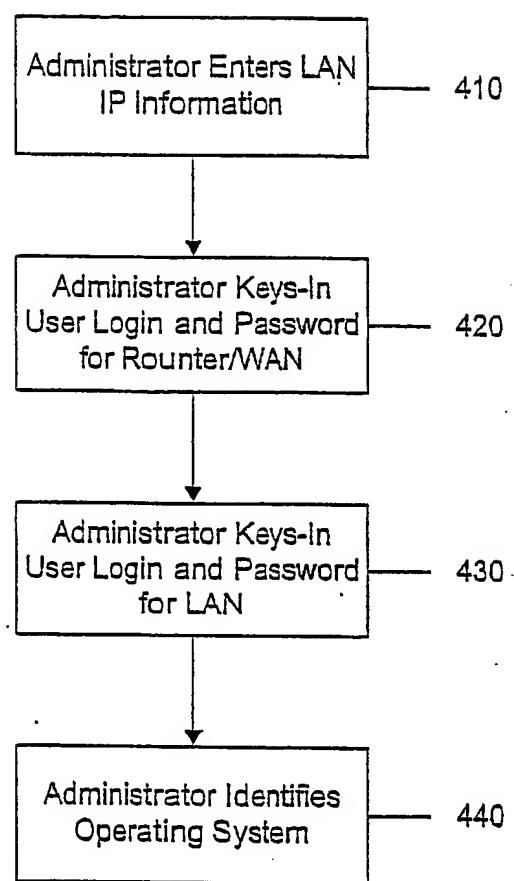
Figure 9

Figure 10



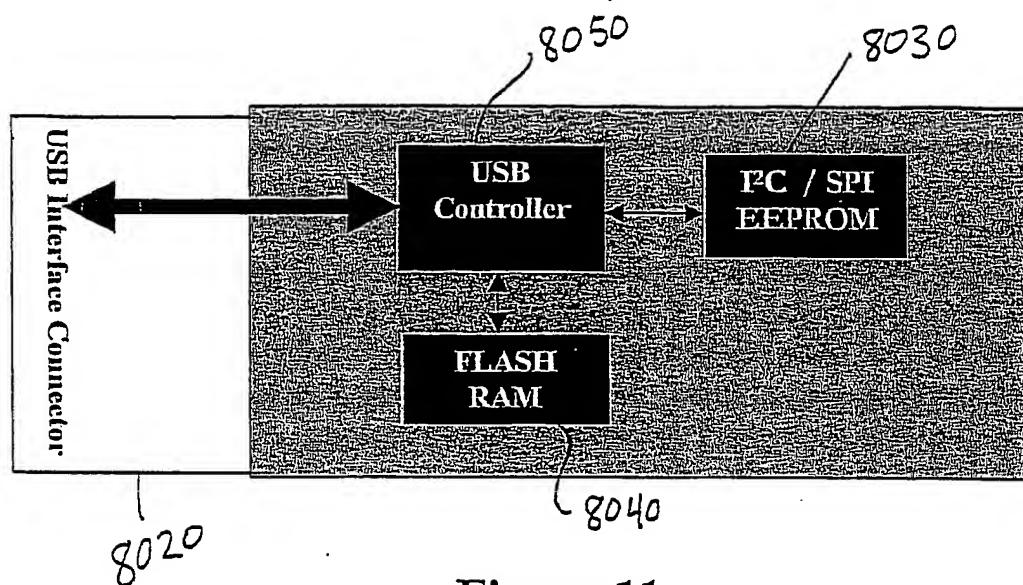


Figure 11



INTERNATIONAL SEARCH REPORT

International application No.
PCT/US03/19407

A. CLASSIFICATION OF SUBJECT MATTER

IPC(7) :G06F 15/177, 13/00
US CL :709/222; 710/104

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 709/202, 220, 221, 222; 710/8, 104; 713/200, 201, 202

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

Please See Extra Sheet.

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 6,108,420 A (LAROSE et al.) 22 AUGUST 2000 ABSTRACT, Column 3, Lines 5-56, Column 5, Lines 33-60, Column 6, Lines 55-67	1-48
Y	US 6,295,556 B1 (FALCON et al.) 25 SEPTEMBER 2001 ABSTRACT, Column 1, Line 50 THROUGH Column 2, Line 24, Column 3, Line 18 through Column 7, Line 67	1-48
A, P	US 6,493,751 B1 (TATE et al) 10 DECEMBER 2002 Entire document	1-48
A	US 6,324,690 B1 (LUU) 27 NOVEMBER 2001 Entire document	1-48

Further documents are listed in the continuation of Box C. See patent family annex.

* Special categories of cited documents:	"T"	later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A" document defining the general state of the art which is not considered to be of particular relevance	"X"	document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"E" earlier document published on or after the international filing date	"Y"	document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"&"	document member of the same patent family
"O" document referring to an oral disclosure, use, exhibition or other means		
"P" document published prior to the international filing date but later than the priority date claimed		

Date of the actual completion of the international search	Date of mailing of the international search report
08 AUGUST 2003	26 AUG 2003

Name and mailing address of the ISA/US Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20251 Facsimile No. (703) 305-3930	Authorized officer MARC THOMPSON Telephone No. (703) 305-3900
--	---

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US09/19407

B. FIELDS SEARCHED

Electronic data bases consulted (Name of data base and where practicable terms used):

EAST USPAT/EPO text search; IEEE, ACM non-patent literature text search
Terms: installation/configuration program, VPN configuration, netowkr computer configuration, WINS, RAS, USB,
WAN profile, gateway